THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 60, No. 8, October 1981
Printed in U.S.A.

Servicing and Real-Time Control of Networks With Dynamic Routing

By G. R. ASH, A. H. KAFKER, and K. R. KRISHNAN

(Manuscript received April 3, 1981)

The design of a network for dynamic routing is made using the forecasted network loads. Load uncertainties arising from errors in the forecast and from daily variations in network load give rise to reserve or idle network capacity not immediately needed by current network demands. The reserve capacity can be reduced by the use of more flexible dynamic routing methods, which allow routing flexibility to help control network flow under load uncertainties. We illustrate techniques for changing network routing patterns in planned and demand servicing to counteract the effects of forecast errors. Included in the benefits are a reduction in both reserve capacity, estimated to be about 5 percent of network first cost, and in trunk rearrangements. We also present call-by-call simulation results for real-time routing enhancements to the basic routing algorithms. The real-time routing algorithms use dynamic trunk reservation techniques, and the simulation results illustrate the improvement in network efficiency and performance under normal daily load variations, network overloads, and network failures.

I. INTRODUCTION AND SUMMARY

Dynamic routing is a new routing system that uses nonhierarchical, time-variable routing patterns to minimize network cost, as opposed to present routing rules that are time-fixed. The term "dynamic" frequently suggests an extensive real-time search for the optimal routing patterns. Real-time, traffic-sensitive routing is indeed the limiting case of time-variable routing, but, as we will see, the degree of load uncertainty determines the needed extent for this "true dynamic routing."

A companion article describes algorithms for designing minimum cost traffic networks using dynamic routing.[1] These design procedures

were investigated under idealized conditions of perfectly known loads: the effects of errors in predicting the loads and other load uncertainties were ignored.

If the future loads on the network were completely known, then it would be sufficient to design the minimum-cost network to meet these loads—for example, by applying the procedure described in Ref. 1. In actuality, various categories of load uncertainty are present that necessitate a somewhat different strategy in building the network.

Network demands are continually growing and shifting which means we must forecast, design, and plan the required capacity far enough in advance (approximately 1 to 2 years) to meet the load. Of course, these forecasts are subject to error and the recognition of this error influences our planning strategy in various ways. The goal is to provide sufficient capacity to meet the expected load on the network. In planned servicing, the servicer plans the network based on the forecast loads and the trunks already in place. Consideration of the in-service trunks results in a disconnect policy that may leave capacity in place even though it is not called for by the design.

There are, however, economic and service implications of the planned servicing policy. Insufficient capacity means that occasionally trunks must be connected on short notice if the network load requires it. This process is called demand servicing. For many reasons it is desirable to minimize the level of demand servicing. There is a trade-off between reserve capacity and demand servicing which we explore in this paper. The algorithms described in Ref. 1 are enhanced to provide efficient planned servicing and demand servicing procedures. Using small network models, we find that these algorithms provide a potential 5 percent reduction in reserve capacity, while retaining a low level of demand servicing.

Uncertain variations in the instantaneous network loads also imply that capacity is never perfectly matched to the demand. Loads on the network shift from hour to hour and from day to day, and some amount of reserve capacity is almost always present. Hence, there is an opportunity to seek out this capacity in real time. We discuss a real-time routing algorithm that finds and uses idle network capacity to satisfy current loads. The procedure is a straightforward enhancement to the planned dynamic routing patterns, and small models predict that network blocking probability is reduced from about 0.0025 to 0.0006.

II. DYNAMIC ROUTING BACKGROUND

2.1 Routing method

The proposed routing method, illustrated in Fig. 1, is called two-link dynamic routing with crankback.

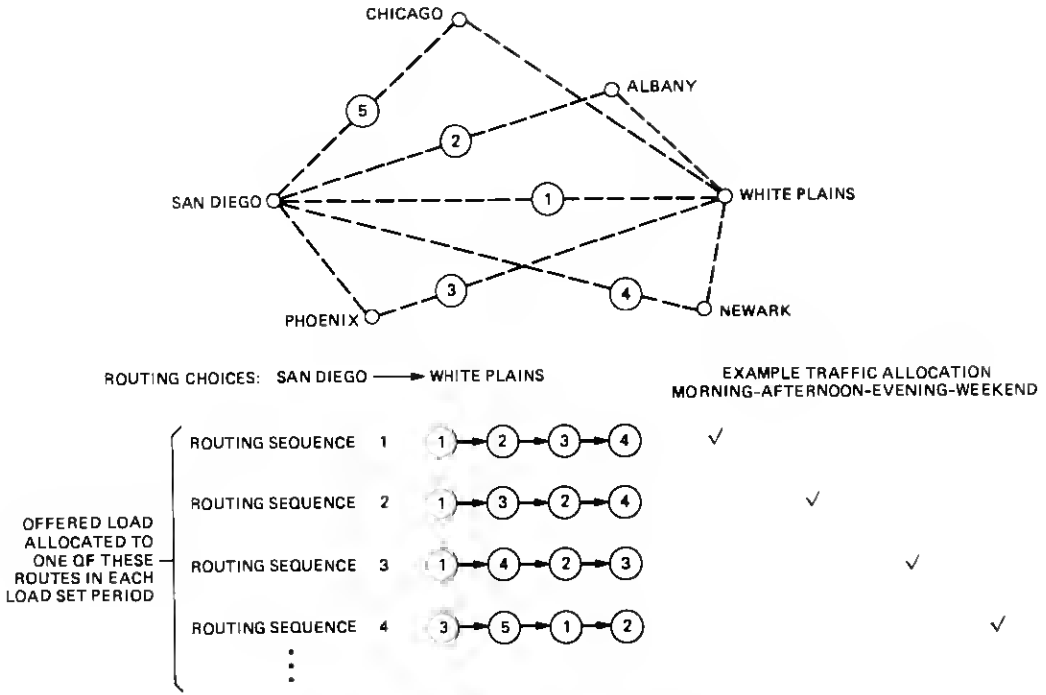

Fig. 1—Two-link dynamic routing with crankback.

The strategy was developed in the companion article and it capitalizes on two factors:

(*i*) Selection of minimum cost paths between the originating and terminating nodes, and

(*ii*) Designing optimal, time-varying routing patterns to achieve minimum cost trunking by capitalizing on noncoincident network busy periods.

The dynamic, or time-varying, nature of the routing scheme is achieved by introducing several route choices. The routes consist of different orderings of the available paths (in Fig. 1, five paths). Each path consists of one or at most two links or trunk groups in tandem. The originating office [San Diego, Ca. (SNDG)] in Fig. 1 retains control over a dynamically routed call until it is either completed to its destination or blocked from the network. A call overflowing the second leg of a two-link connection [e.g., the Albany, N.Y.-White Plains, N.Y. (ALBY-WHPL) link of the SNDG-ALBY-WHPL path] is returned to SNDG, the originating office, for possible further alternate routing. Control is returned by using the common-channel interoffice signaling (CCIS) crankback signal sent from the via-node to SNDG.

Each of four routing sequences illustrated in Fig. 1 uses a different order of the five paths. Each routing sequence results in a different allocation of link flows, but all satisfy the point-to-point grade-of-service requirement. Allocating traffic to the optimum route choice during each load-set-period leads to design benefits due to the noncoincidence of loads. This route selection changes with time as shown in the columns on the right, thus, it is dynamic. The example shown indicates that in the morning the routing strategy is to offer the SNDG-WHPL traffic to routing sequence number one (starting with the direct trunk group to WHPL overflowing to the two-link connection through ALBY) and, in the afternoon, to routing sequence number two [overflowing to the two-link connection through Phoenix, Az. (PHNX)]. In the evening, routing sequence number three is used.

2.2 Design algorithm

The basic steps of the dynamic nonhierarchical routing (DNHR) design algorithm are shown in Fig. 2.[1] The algorithm combines several techniques for achieving network savings into a single, unified approach.[2-5] The steps of the design algorithm illustrated in Fig. 2 show that it is an iterative technique consisting of a router, an engineering module, and an update module. See Ref. 1 for a more complete discussion of this algorithm.

III. SOURCES OF LOAD UNCERTAINTY AND CONTROL OF ROUTING

The telephone network is designed on the basis of forecasted loads,

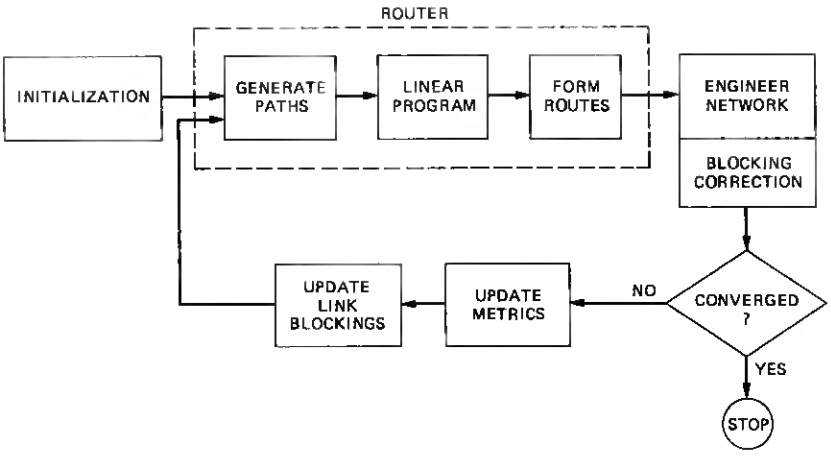

Fig. 2—Unified algorithm block diagram.

since the network capacity must be available before the loads occur. Errors in the forecast lead to uncertainty about the actual loads that will occur. In addition, each forecasted load is actually a mean load about which there occurs a day-to-day variation, characterized by a gamma distribution with one of three levels of variance.[6] Even if the forecast mean loads are correct, the actual realized loads exhibit a random fluctuation from day to day. Hence, there are two sources of load uncertainty: forecast error and day-to-day variation. Earlier studies have established that each of these sources of uncertainty requires the network to be augmented in order to maintain the grade of service.[7,8]

Control over network capacity is divided between planned servicing and demand servicing. Planned servicing is an annual process that determines where network capacity is needed to meet the future demand. Major inputs to planned servicing are the load forecast, which is subject to error, and the existing network. Trunk disconnects are determined in planned servicing when the forecast predicts declining or shifting loads, and the servicer is reasonably sure the trunks will not be needed in the next 1 to 2 years. This procedure reflects some reluctance to disconnect trunks, and results in a certain amount of reserve capacity being left in the network. Planned servicing drives the bulk of trunking activity, which is scheduled over the next yearly interval.

On occasion, the planned servicing strategy underprovides trunks at some point in the network, again, because of forecast errors, and the servicer must respond quickly to restore service. The process of correcting for these forecast errors is called demand servicing.[9] When some trunk groups are found to be overloaded as a result of the actual

loads being larger than their forecast values, additional trunks are provided to restore the grade of service to the required value. Trunks will not usually be disconnected in demand servicing, and, as a result, the process leaves the network with a certain additional amount of reserve or idle capacity even when the forecast error is unbiased.[7]

The effects of day-to-day variation, unlike those of forecast error, are taken into account in the initial design of the network[8] and arise from the nonlinear relation between trunk-group load and blocking. When the load on a trunk-group fluctuates about a mean value, because of day-to-day variation, the mean blocking is higher than the blocking produced by the mean load. Therefore, additional capacity is provided to maintain the grade of service in the presence of day-to-day load variation.

The question is: To what extent can the capacity augmentation required by the uncertainties be reduced by dynamically controlling the routing patterns to meet the realized loads? A given realization of the loads can be expected to yield some parcel (point-to-point) loads which are higher than average and others which are lower. While part of the network is overloaded, another part might be underloaded. If the routing pattern can be adjusted to use the idle capacity of the underloaded portion of the network, the required capacity augmentation might be reduced.

Figure 3 shows the relation among the three levels of routing control, in which the design of trunk-group sizes and routing patterns of the network is viewed as feedback process. The outermost loop represents planned servicing in which link sizes and routing are planned approximately once a year. The next inner loop represents demand servicing which responds to service problems arising from unforecasted demand, at approximately one- to four-week intervals. From measurements of realized loads and blocking, the demand servicing algorithm determines augmentations to the link sizes and modifications to the routing patterns to correct for errors in the load forecast from which the design was made. The inner-most loop represents real-time routing in which only routing modifications are possible. This final level of routing control must deal with:

(*i*) Day-to-day load variations,

(*ii*) The effects of unforecasted demand, until the needed capacity augmentations can be made at the next demand servicing, and

(*iii*) Network management under overload and failure conditions.

IV. PLANNED AND DEMAND SERVICING TECHNIQUES

The flow diagram of Fig. 4 illustrates designing a network on the basis of forecast loads. Planned servicing accounts for both the current network and the forecast loads in planning network changes, and then

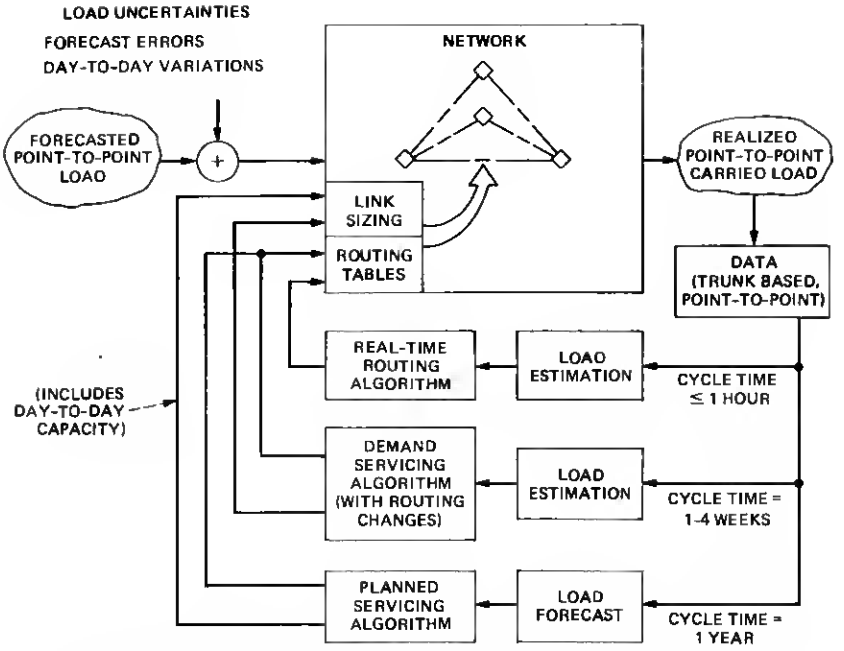

Fig. 3—Planned servicing, demand servicing, and real-time control as interacting feedback loops around the network.

demand servicing makes routing and trunking adjustments, if network performance under the realized loads becomes unacceptable because of errors in the forecast.

As discussed earlier, the planned servicing strategy tries to minimize reserve capacity, while maintaining an acceptable level of demand servicing. The model in Fig. 4 assumes that planned servicing is an annual process which predicts the required network capacity to meet the future demand. It considers both the demand forecast, which is subject to error, and the existing network. In dealing with forecast errors, planned servicing attempts to provide sufficient network capacity to meet these demands with a minimum of demand servicing. Our model assumes that the trunk network resulting from planned servicing is implemented immediately and then demand servicing is invoked to restore network service when shortages are detected.

4.1 Planned servicing methods

4.1.1 Conventional planned servicing

In the current hierarchical network, planned servicing begins by comparing the existing network with a network designed for the forecast loads. The design is made without reference to the existing

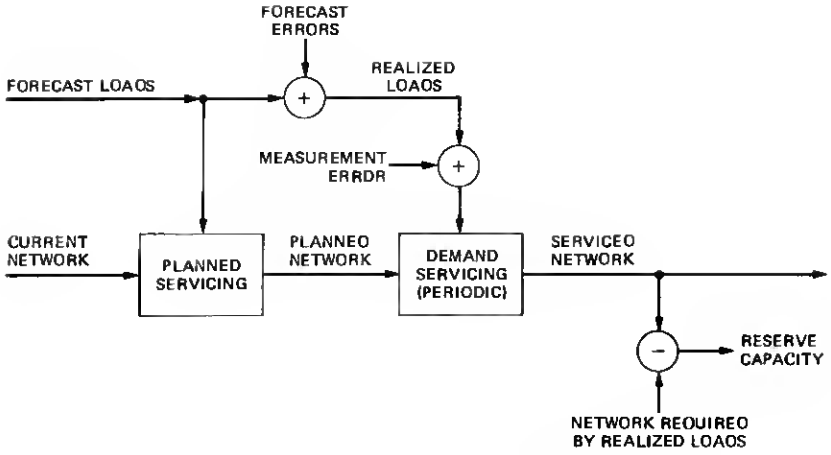

Fig. 4—Model of the planned and demand servicing process.

trunk group sizes. When the forecasting system calls for additional trunks on a group, the augments are usually implemented. If the forecasting system calls for fewer trunks, a disconnect policy is invoked to decide whether trunks should be disconnected, and, as discussed earlier, this policy reflects a degree of reluctance to disconnect trunks. The conventional method of planned servicing, which tries to guide the existing network towards an ideal network designed for the forecast loads, could be used in the DNHR network also. If there are substantial differences in the structure of the forecast network from one year to the next, conventional planned servicing, combined with the reluctance to disconnect trunks, might produce augmentation that could be avoided by better use of existing capacity.

4.1.2 Incremental planned servicing

Given the reluctance to disconnect trunks, it seems reasonable that planned servicing should design a network considering the trunking which is in place. This can be done using incremental planned servicing where, instead of designing an ideal network from the beginning, we design a minimum-cost augmentation of the existing network to meet the forecast loads. Slight modifications to the traffic routing and engineering blocks in the design algorithm acomplish this change, which allows us to use given initial link capacities as lower bounds on the designed link capacities. In effect, the algorithm takes into account the reluctance to remove trunks. This procedure limits the trunk augments to those required to meet the forecast loads and, thus, achieves a lower reserve capacity.

4.1.3 Generalized incremental planned servicing

Incremental planned servicing designs a minimum-cost augmentation to the existing network. We generalize this procedure to set minimum trunk group sizes and to allow trunk disconnects. This is done by deriving, for each group, a lower threshold and an upper threshold for its size, and using these thresholds to make initial adjustments to the group size (as described below) prior to incremental planned servicing.

The size thresholds for a group are based upon the forecast loads of the corresponding direct parcel, and are determined by choosing a minimum value r_{min} and a maximum value r_{max} for the ratio $r \triangleq$ (trunk group size)/(direct parcel load). The lower (reserve) threshold for the group size is based on the forecast peak load of the direct parcel in the next year and corresponds to the ratio r_{min}. The upper (disconnect) threshold is based on the forecast peak load of the direct parcel over the next two years, with some allowance for forecast error, and it corresponds to the ratio r_{max}.

The limits r_{min} and r_{max} were chosen by examining the range of values of the ratio in a typical DNHR network design. In general, the ratio has a smaller spread of values for large parcels than for small parcels. With large parcel loads, the corresponding group can be quite efficient carrying just the direct parcel; hence, its size to a large extent depends just on the direct parcel. For small parcel loads, however, the group size is less dependent on the direct parcel load and is more influenced by the alternate routing parcels carried on that group; hence, the ratio is expected to have a wider range of values.

Table I shows the limits for the ratio (trunk group size)/(direct parcel load), as a function of the direct parcel load.

The lower threshold T_{min} and upper threshold T_{max} for a trunk group are determined in terms of the forecast loads for the corresponding direct parcel:

Table I—Limits on $r = T/L$ = (trunk group size)/(direct parcel load)

Load L (erlangs)	$r_{min}(L)$	$r_{max}(L)$
0–5	0.3	4.5
5–10	0.4	4.5
10–25	0.5	3.0
25–50	0.7	3.0
50–100	0.95	2.5
>100	1.05	1.5

Let L_i = peak forecast load for the direct parcel in year i, $i = 1, 2$.

β_i = forecast uncertainty factor for year i, $i = 1, 2$, introduced to allow for probable error in the load forecast. The results presented were obtained with $\beta_1 = 1.15$, $\beta_2 = 1.3$ corresponding to 0.15 coefficient of variation in the forecast.

Then,

$$T_{\min} \triangleq r_{\min}(\beta_1 L_1) * \beta_1 L_1$$

$$T_{\max} \triangleq \max[r_{\max}(\beta_1 L_1) * \beta_1 L_1,\ r_{\max}(\beta_2 L_2) * \beta_2 L_2],$$

where $r_{\min}$ and $r_{\max}$ are the appropriate limits established for ratio (T/L), such as those in Table I.

With these lower and upper thresholds, we then define an initial size for each group, which depends on its current size as follows:

(*i*) If the current size of a group is between its lower and upper thresholds, its initial size equals its current size.

(*ii*) If the current size of the group is below its lower threshold, its initial size equals the lower threshold.

(*iii*) If the current size of the group is above its upper threshold, its initial size equals the upper threshold.

We use the initial network defined in this manner as the starting network for incremental design (i.e., minimum-cost augmentation) to arrive at the forecast network for each future year, as described in Section 4.2. Comparing the result with the current network, we determine the actual augments and disconnects that must be made to implement the forecast network. Under normal growth conditions, the current trunks are most often used in the initial network, not the upper and lower trunk limits, and the primary effect is to route traffic on the actual trunks in place and, thus, minimize rearrangments.

4.2 Planned servicing algorithm

As noted earlier, the unified algorithm (UA) for DNHR network design is an iterative procedure with four basic steps (Fig. 2): selection of cost-effective traffic paths, optimization of path flows, sizing the trunk groups (engineering) to correspond to the optimum flows, and updating of marginal link costs and optimum link blockings for the next iteration. The proposed procedures for planned and demand servicing involve modifications to the flow optimization and engineering routines of the UA to allow the existing link capacities to be used as lower bounds on the designed link capacities.

We now describe the modifications to the flow optimization and engineering procedures for use in planned servicing.

Let

L = number of links.

H = number of design hours.

$b_i^{\max}$ = maximum permitted blocking on link i.
b_i^h = blocking of link i in hour h, $h = 1, \cdots, H$.
y_i^h = carried load on link i in hour h, $h = 1, \cdots, H$.
a_i = capacity of unaugmented link i, in carried load at blocking $b_i^{\max}$.
Δa_i = capacity augmentation, in carried load, on link i.
M_i = marginal cost of augmentation, in cost per erlang, on link i.

$i = 1, \cdots, L$

4.2.1 Flow optimization

The object is to allocate the traffic flow of each hour among its admissible paths so as to minimize the cost of the required link capacity augmentations. On each link, for a given number of existing trunks and the maximum economic blocking determined from economic considerations,[3] there is a maximum load that can be carried on that link without augmentation; this is the unaugmented initial capacity of that link.

The flow optimization problem for planned servicing is now stated as a linear program in which the decision variables are the flow assignments and the augmentations Δa_k above the existing link capacities a_k (instead of total link capacities as in the design problem described in Ref. 1), and the cost to be minimized is the marginal cost of augmentation

$$\sum_{k=1}^{L} M_k \Delta a_k.$$

This formulation ensures that efficient use is made of existing link capacities, by means of routing changes if needed, before link augmentations are proposed.

4.2.2 Engineering

In engineering we are given the traffic routing and loads and find the needed augmentation to those groups which exceed their maximum permitted blockings. This is accomplished by the following iterative procedure:

(*i*) Begin with assumed link blockings $\hat{b}_i^h$ (e.g., the link blockings in the unaugmented network) subject to $\hat{b}_i^h \leq b_i^{\max}$, $i = 1, \cdots, L$.

(*ii*) Calculate the corresponding carried link loads y_i^h, under the known routing and assumed link blockings.

(*iii*) If for all $h, y_i^h \leq a_i$, the capacity of the unaugmented link at its maximum blocking, then the link needs no augmentation; if $y_i^h > a_i$, the required augmentation Δa_i is determined by engineering the link for load y_i^h at blocking $b_i^{\max}$.

(*iv*) From the link loads y_i^h and link sizes computed in (*ii*) and (*iii*),

we recalculate all the link blockings b_i^h; if $|b_i^h - \hat{b}_i^h|$ is not sufficiently close to zero for all i in all hours h, redefine $\hat{b}_i^h = b_i^h$, $i = 1, \cdots, L$, $h = 1, \cdots, H$, and return to (*ii*).

The marginal link costs and optimum link blockings are, in general, determined for the forecast loads each year during planned servicing, although in some years their values might change little from the previous year.

4.3 Demand servicing methods

Between successive planned servicings, if the realized loads exceed forecast values and cause unacceptable blocking, then quick corrective action, called demand servicing, is needed. In the current hierarchical network, demand servicing is usually limited to trunk group augmentations. However, in the DNHR network, the basic routing patterns are time-variable, and hence, routing modifications can be used in demand servicing to reduce network augmentation. To the extent that routing changes can be substituted for the installation of trunks, rearrangements are also reduced.

Demand servicing consists of three steps:

(*i*) Detecting the need for demand servicing, i.e., determining whether or not all parcels are receiving adequate service,

(*ii*) If servicing is needed, then determining the best combination of routing changes and link augmentations that will restore the desired grade of service at minimum cost of augmentation, and

(*iii*) Implementing the routing changes and link augments. These steps are discussed in more detail below.

4.3.1 Detection of service problems

Point-to-point blocking measurements are needed in the DNHR network to determine the level of service being provided and, thus, to detect the existence of service problems. Because of measurement errors and day-to-day traffic variations, such blocking measurements will have an inherent statistical variability which must be allowed for by establishing acceptable bands for the measured blockings.

4.3.2 Demand servicing algorithm

The need here is for a simple procedure to determine the corrective action required; there is no attempt to redesign the whole network, or to disconnect trunks, if the network is found to be overprovided for the realized loads. We use the flow-optimization routine to determine the optimum traffic routing for the realized loads. Using this optimum routing, we use the engineering routine to determine the link augmentations required to limit link blockings to their maximum permitted values. If some parcel blockings remain higher tban desired after this

step, we invoke the blocking correction procedure discussed in Ref. 1 to correct the problem. Thus, a procedure similar to the incremental planned servicing algorithm is used to determine the required changes in demand servicing.

4.3.3 Implementing routing changes

Using demand servicing with routing changes, the network routing might change very frequently: possibly at every demand servicing interval. Manual administration of such frequent routing changes in the network would be unmanageable, and a substantial degree of automation will be required in implementing the routing changes. This suggests a network routing data base which would receive routing revisions from the output of demand servicing. With such a data base and an automatic update system, the administration of routing changes in demand servicing appears quite feasible.

4.4 Servicing results

The servicing model in Fig. 4 was used to simulate the servicing process on a 28-node network (Fig. 5) to determine the effectiveness of the proposed planned and demand servicing procedures. The network, starting from a design for the first year's forecast loads, was taken through 10 years of the servicing process, each iteration consisting of a forecast at the beginning of the year, followed by a demand servicing during the year. The forecast parcel loads grew at a 5-percent annual rate. To simulate forecast error, the realized parcel loads in each year were assumed to be normally distributed about the forecast loads, with a 15-percent coefficient of variation.

The following three schemes were compared:

Scheme A—Conventional planned servicing and demand servicing with routing changes.

Scheme B—Incremental planned servicing and demand servicing with routing changes.

Scheme C—Generalized incremental planned servicing and demand servicing with routing changes.

In schemes A and B, no disconnects were allowed in planned servicing in order to simulate complete reluctance to disconnect trunks. In all three schemes, no disconnects were allowed in demand servicing.

Figure 6 shows the evolution of network reserve capacity for the three servicing schemes, measured by the percentage difference in cost between the realized network and an ideal network designed for the realized loads. Figure 7 shows the cumulative demand servicing trunk augments for the three schemes as percentages of the number of trunks in the starting network. Figure 8 shows, for each scheme, the level of demand servicing in each year, as measured by the trunk augments in demand servicing as a percentage of trunks in the realized network.

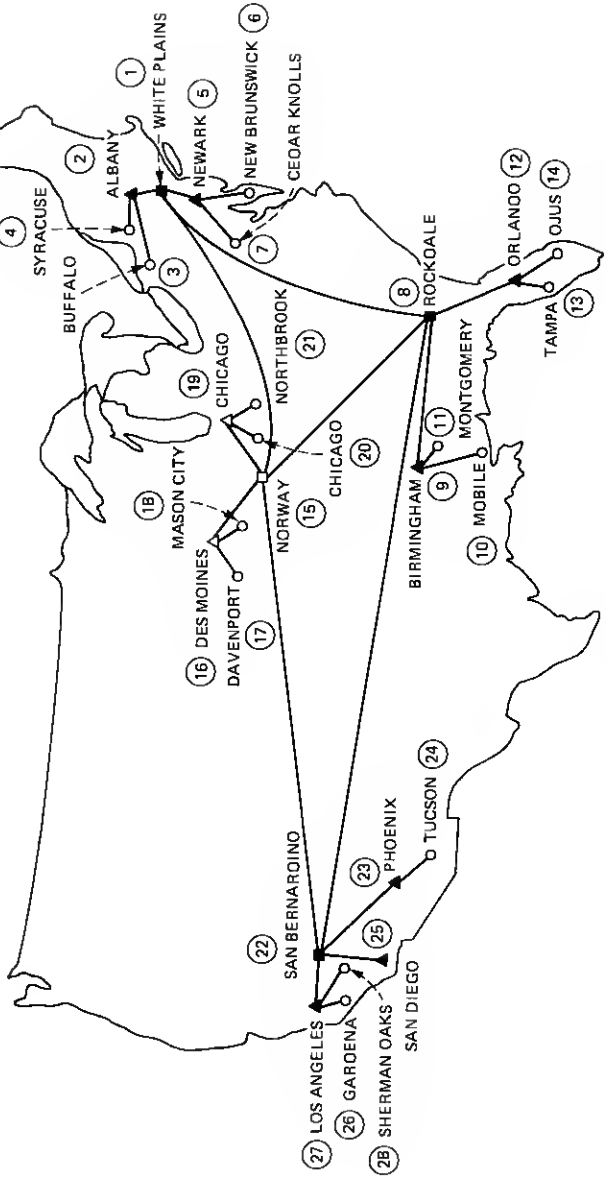

Fig. 5—Intercity network model for 28-node network. (Dark nodes indicate 10-node model.)

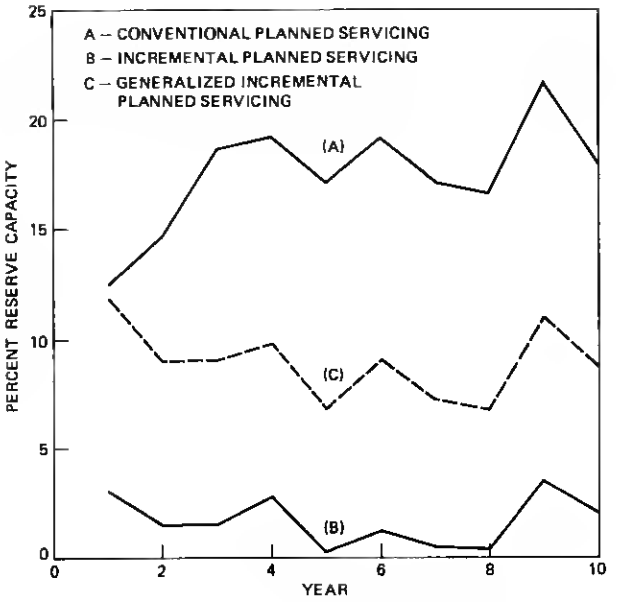

Fig. 6—Evolution of network reserve capacity.

We note from Figs. 6 and 7 that incremental planned servicing (scheme B), as expected, achieves a lower reserve capacity than conventional planned servicing (scheme A) but requires more demand servicing augments in response to forecast error. The generalized incremental planned servicing method (scheme C) falls between the other two both in reserve capacity and in demand servicing. Compared to conventional planned servicing, it achieves a striking reduction in reserve capacity for a modest increase in demand servicing.

Figure 8 shows that, in all three schemes, demand servicing rearrangements in each year are in the range of 1 to 4 percent of the trunks in the network, a level that is quite favorable in comparison with the demand servicing level in the current hierarchical network (estimated to be about 10 percent of the trunks in the network).

Figure 9 shows the cumulative *total* rearrangements (consisting of augments and/or disconnects in planned servicing and augments in demand servicing) for the three schemes as percentages of the number of trunks in the starting network. We note that when the total trunk changes occurring in planned and demand servicing are considered, incremental planned servicing (scheme B) produces the fewest rearrangements and conventional planned servicing the most, with generalized incremental planned servicing falling in between the other two. However, in general, more time is available for implementing planned servicing rearrangements than demand servicing rearrangements,

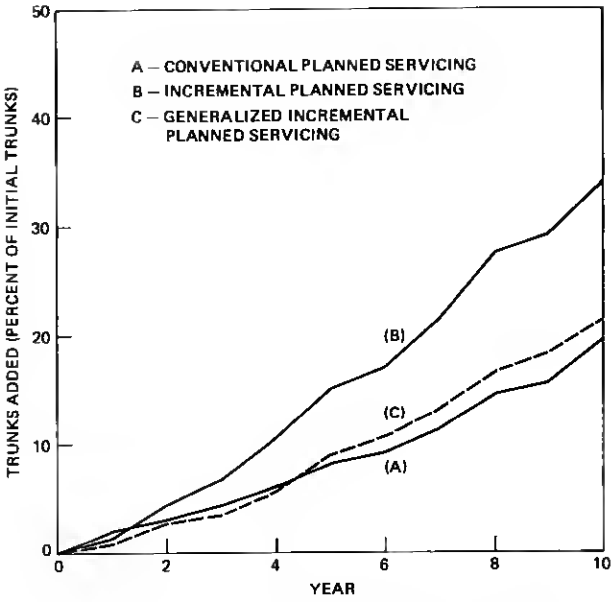

Fig. 7—Cumulative demand servicing rearrangements.

which are called for on short notice, to correct existing service problems. It is therefore likely that demand servicing rearrangements are more expensive than planned servicing rearrangements. Taking this into account, we may expect that the administrative cost of rearrangements is smaller with generalized incremental planned servicing than with just incremental planned servicing.

Figures 6 and 7 have pointed to the trade-off that exists between reserve capacity and demand servicing rearrangements, and generalized incremental planned servicing has been proposed as a method of securing the desired trade-off between these two aspects. For example, by multiplying the factors $r_{\min}$ in Table I by a factor $\alpha \geq 0$, we can parametrize the resulting levels of reserve capacity and demand servicing. The value $\alpha = 1$ corresponds to curve C in Figs. 6 to 9; $\alpha < 1$ results in lower reserve capacity and increased demand servicing, while $\alpha > 1$ leads to higher reserve capacity and reduced demand servicing.

Figure 10 is a curve of average reserve capacity versus the average level of demand servicing (average over 10 years) in scheme C, with α as the parameter. For comparison, the two points corresponding to schemes A and B, respectively, are also plotted. Point B is almost the same as the limiting case $\alpha = 0$, while point A lies above the curve, showing that a more favorable trade-off can be obtained with scheme C than with A. This curve is a quantitative expression of the trade-off between reserve capacity and demand servicing.

We conclude that generalized incremental planned servicing, combined with demand servicing with routing changes, is an efficient method of controlling reserve capacity and the level of demand servicing in the network. On the basis of the results presented in this paper, a reserve capacity level of about 7 to 10 percent appears suitable for the assumed 15-percent coefficient of variation of load forecast error. We have not presented a direct comparison between demand servicing with and without routing changes. Such a comparison has been made on a 10-node subset of the 28-node network (Fig. 5). The results show that routing changes in demand servicing reduce network cost by about 2 to 3 percent and demand servicing rearrangements by about 15 percent.

V. REAL-TIME ROUTING CONTROL

Planned servicing and demand servicing will account for known, systematic load variations including unforecasted demand in the planned routing patterns and trunk group sizes. The only routing decisions necessary in real time involve conditions that also become known in real time: day-to-day load variations, network failures, and network overloads.

The day-to-day component of load variation is not systematic and/or easily predictable because it involves daily load shifts which are

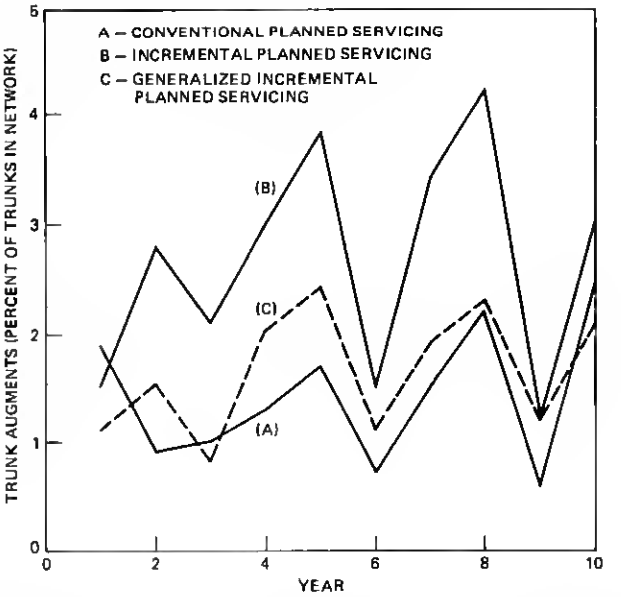

Fig. 8—Demand servicing level. Trunk augments in demand servicing as a percentage of trunks in network.

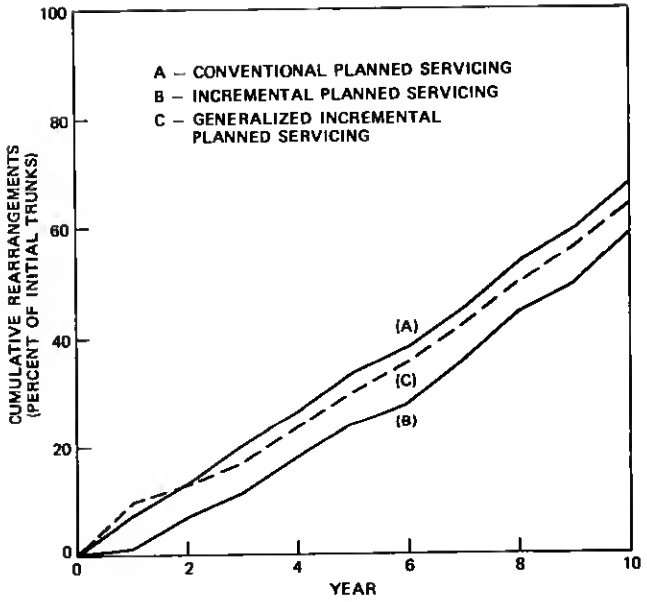

Fig. 9—Cumulative trunk rearrangements. Augments plus disconnects.

essentially random from one day to the next.* Reasonably accurate load patterns can be predicted months in advance; the unforecasted demands can then be identified and corrected over a period of a few weeks (as the loads develop), but daily load variations must ultimately be identified in real time.

The network design will size the network to accommodate all expected load patterns including day-to-day load variations. Sizing the network for day-to-day variations will guarantee that some capacity will stand idle at least some of the days. If planned routing patterns were totally preprogrammed, no advantage could be taken of temporarily idle network capacity to complete calls that might otherwise be blocked. For this reason, a method of extending routing patterns beyond the preprogrammed sequence to include real-time decisions was devised.

Real-time routing can be used to improve network service. Service improvement is significant even with relatively simple procedures—the improvement can also be equated to an equivalent trunk cost savings of about 2 to 3 percent or improved network service with a higher overall completion rate. Real-time dynamic routing should also

* It is known that some daily variations are systematic (e.g., Monday is usually higher than Tuesday). However, in the present environment, these known changes are ignored and lumped into the stochastic model.

improve network performance somewhat in the event of network failures, especially when some amount of reserve capacity is available for redirecting traffic flows from their usual patterns.

5.1 Real-time routing method

A relatively simple real-time procedure is investigated which is a natural extension of the two-link routing procedure being proposed. The method appends to each sequence of two-link paths, engineered by the design algorithm for the expected network load, additional two-link (real-time) paths to be used only after the normal sequence is exhausted and only when idle capacity is available.

Dynamic trunk reservation is used to help recognize idle network capacity. Access to trunks on a particular trunk group is allowed only after a specified number of trunks—the reservation level—is available. Reservation guarantees that capacity is truly idle and accessing it will produce minimal interference with normal traffic.

The selection of real-time paths for each point-to-point pair can be done as a natural extension of the design and servicing algorithms. These algorithms recognize noncoincidence factors and can identify groups that are expected to have slack capacity at a particular time. Causes of slack capacity include forecast errors, the disconnect policy, and reserve capacity from modular engineering. The candidate real-time paths would be selected off-line from the list of paths generated

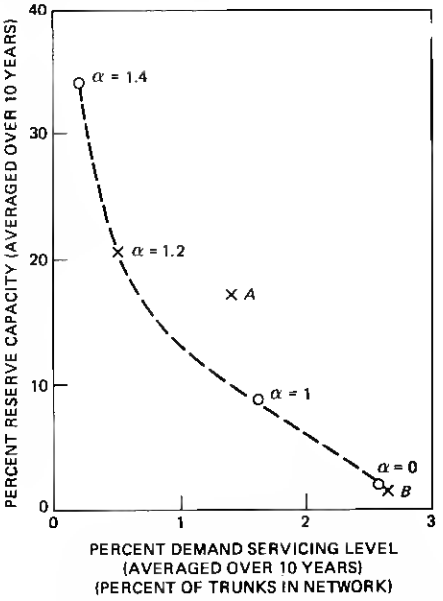

Fig. 10—Trade-off between reserve capacity and demand servicing level.

by the algorithms. This list contains a large number of potential paths to be used in forming the planned routes. The real-time paths are chosen from those paths not already used as part of the planned route. It should also be noted that the larger the allowed number of real-time paths, the lower the blocking for an individual point-to-point pair. However, there are administrative costs, storage limitations, and real-time penalties restricting the number of allowed paths.

Three means of routing calls among the real-time paths were studied: a sequential method identical to the planned routing patterns described in Section 2.1, and two cyclic routing methods in which the real-time calls are rotated among the real-time paths analogous to the current automatic-out-of-chain-routing (AOOCR) method (described in Section 5.2). It was found that the sequential method provided equal or better performance than the others and, hence, is preferred because of its ease of implementation.

5.2 Alternative methods of real-time routing

There are several possible methods for implementing real-time routing which use call-by-call routing decisions. Methods under active study are discussed in this section.

5.2.1 Automatic-out-of-chain routing

The No. 4 ESS provides an expansive control AOOCR.[10] With AOOCR, overflow from a final trunk group, which is presently routed to a reorder announcement or manually rerouted, is sent to an out-of-chain route where it will attempt to complete. Up to seven out-of-chain routes can be identified for each final group, and the No. 4 ESS will spread the overflow traffic uniformly over these routes, as capacity is available. This is accomplished by using a cyclic routing method which, for each call, tries the path following the previous path attempted. All out-of-chain traffic is accompanied by a CCIS traveling class mark so that it receives special treatment at the via-office to prevent shuttling and will be turned back unless the via-route has available capacity. If a call fails to find a free trunk leaving the via-node, the call is blocked at the originating office and the particular out-of-chain path is turned off from further attempts for a period of about 30 seconds.

5.2.2 Learning automata

Another decentralized routing scheme involves the use of learning automata.[11-13] A learning automaton is a machine (or an algorithm) whose actions are constantly modified by feedback from its environment. The updating procedures used to modify the actions of the different types of automata determine their learning characteristics. One example is the L_{R-I} automaton (linear reward-inaction). In this

scheme, if a particular routing choice, or action, gets a positive response from the environment (i.e., the call is completed), the probability of choosing this action for subsequent calls is increased. However, if a negative response is received, the action probabilities are not modified. A detailed mathematical model for the L_{R-I} automaton can be found in Ref. 12.

Models for the sample mean (M) automaton and the linear reward-penalty (L_{R-I}) automaton have also been developed for possible use in the telephone network. Simulation studies with simplified networks show that these automata perform better than a fixed (hierarchical) routing strategy.[11,13]

5.2.3 Centralized real-time routing

A centralized routing system has been investigated by Bell Northern Research.[14] In this implementation, the selection of candidate paths at each switch is recalculated every two seconds. The path selection is done by a central routing processor, based on the busy-idle status of all trunks in the network. This system was field tested for four months on nine switching systems in Toronto. A computer analysis of actual call-by-call demand on the network showed that advanced routing provides more uniform and better service characteristics than the hierarchy during overloads.

The preceding have been examples of different techniques of real-time routing. However, the decentralized real-time routing method investigated bere seems to be a practical method for large networks at the present time.

5.3 Simulation results

In this section, we summarize the results of a call-by-call simulation using the 28-node intercity network model (Fig. 5). The simulations selected were guided by results from small analytic models, and the simulation results were obtained using the 10 a.m. (EST) busy-hour load and routing.

The call-by-call simulation model assumed transparent nodes; that is, no queuing or blocking was modeled for the switching systems in the network. Poisson arrivals were used to model originating calls together with an exponential holding time distribution having a mean of five minutes. Day-to-day variations were modeled with a gamma distribution with a variance equal to $0.13a^{\phi}$, with $\phi = 1.5$ to model low daily variations. The parameter "a" represents the point-to-point offered load in erlangs. Reserve capacity was modeled by using a uniform distribution on the trunk group size centered about a 7-percent average reserve capacity and varied uniformly between 3 and 11

percent. It is taken as a typical level of reserve capacity for the dynamic routing network.

5.3.1 Results for day-to-day variations

Table II illustrates the performance of the real-time routing scheme in terms of three network performance criteria used in the simulation:

(*i*) Average network blocking—indicates the effectiveness of completing calls.

(*ii*) Maximum parcel blocking—indicates the interference of real-time calls with traffic using the planned route.

(*iii*) Crankbacks per originating call and machine attempts per originating call—indicate the switching and signaling effort to complete real-time calls. Total crankback attempts are counted; machine attempts include originating, terminating, crankback (counted as one full attempt), and tandem completing.

Notice that real-time routing has little impact on total machine attempts, and that network blocking with real-time routing is reduced by about 75 percent. Figure 11 illustrates the improvement in blocking performance with real-time routing as a function of reserve capacity. We see that its effectiveness increases as reserve capacity increases, which is expected. This improvement in network service will translate into greater revenues by providing a higher network completion rate.

5.3.2 Results for natwork overload and failure

Real-time routing should not degrade network performance under overload and failure conditions. The simulation results verify that real-time routing does not degrade average network blocking or individual parcel blockings under general and focused overload conditions. However, control is needed to limit the generation of crankback messages under these conditions.

A link failure of the Los Angeles, Ca.-Newark, N.J. (LSAN-NWRK) link (23 trunks) was simulated with the results shown in Table III.

These results also show that the average network blocking improves using real-time routing with a slight increase in switching effort. Comparing Tables II and III, we note that under a link failure the

Table II—Performance of real-time routing (low day-to-day variations—7 percent reserve capacity)

Real-time Routing Used?	Avg. Network Blocking	Max. Parcel Blocking	Crankbacks per Orig. Call	Attempts per Orig. Call
No	0.00249	0.017	0.0207	2.17
Yes	0.00058	0.009	0.0230	2.17

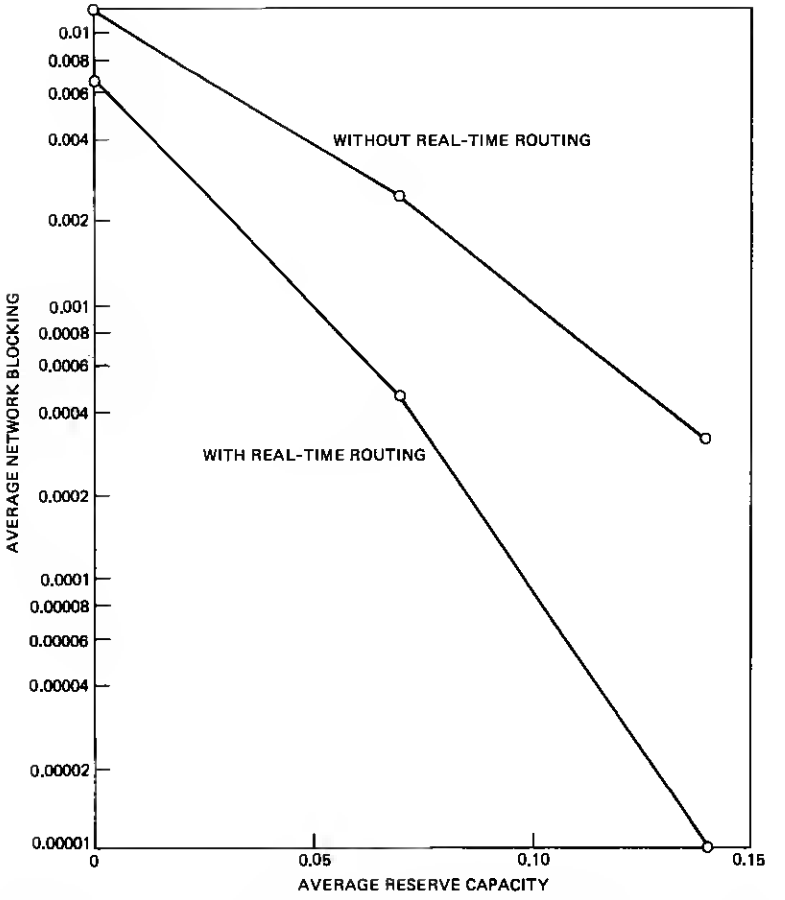

Fig. 11—Average network blocking versus average reserve capacity (low day-to-day variations).

average network blocking increases slightly, but that real-time routing maintains the maximum parcel blocking at the same level.

A node failure of the NWRK node was simulated with the results shown in Table IV. The high blocking parcel in this case was the ALBY-WHPL parcel. Newark, N.J. was normally a via point for this parcel. Additional real-time calls using the ALBY-WHPL link also contributed to the higher parcel blocking. However, it is significant that the overall network blocking improved when real-time routing was applied. All calls destined for NWRK overflowed all the planned and real-time paths causing a large number of real-time path attempts, plus crankback messages. Normally, automatic network management controls would cancel such attempts to alleviate this situation. Another interesting

Table III—Performance under link failure (LSAN-NWRK failure—7 percent reserve capacity)

Real-time Routing Used?	Avg. Network Blocking	Max. Parcel Blocking	Crankbacks per Orig. Call	Attempts per Orig. Call
No	0.00264	0.023	0.0258	2.17
Yes	0.00062	0.009	0.0291	2.18

Table IV—Performance under node failure (NWRK node failed—7 percent reserve capacity)

Real-time Routing Used?	Avg. Network Blocking*	Max. Parcel Blocking*	Crank-backs per Orig. Call	Attempts per Orig. Call
No	0.00819	0.082	0.183	2.19
Yes	0.00099	0.089	0.187	2.20

* Excluding traffic to the NWRK node.

phenomenon is that most parcels not normally using NWRK as a via point achieved better than normal service. This occurred because the trunk groups normally carrying NWRK traffic are relatively free (NWRK traffic cannot complete). Hence, other parcels can make use of these relatively lightly loaded groups to achieve better than normal service.

VI. ACKNOWLEDGMENTS

We thank A. H. Westreich who did the initial work on computer implementation of the servicing model. Thanks are also in order for L. T. Nguyen, who carried out many of the subsequent computer simulations.

REFERENCES

1. G. R. Ash, R. H. Cardwell, and R. P. Murray, "Design and Optimization of Networks with Dynamic Routing," B.S.T.J., this issue.
2. M. Eisenberg, "Engineering Traffic Networks for More than One Busy Hour," B.S.T.J., *56*, No. 1 (January 1977), pp. 1–20.
3. C. J. Truitt, "Traffic Engineering Techniques for Determining Trunk Requirements in Alternate Routed Networks," B.S.T.J., *33*, No. 2 (March, 1954), pp. 277–302.
4. B. Yaged, Jr., Long Range Planning for Communications Networks, Polytechnic Institute of Brooklyn, Ph.D Thesis, 1971.
5. J. E. Knepley, "Minimum Cost Design for Circuit Switched Networks," Technical Note Numbers 36–73, Defense Communications Engineering Center, System Engineering Facility, Reston, Virginia, July, 1973.
6. R. I. Wilkinson, "A Study of Load and Service Variations In Toll Alternate Route Systems," Proc. 2nd Int. Teletraffic Congress, The Hague, July, 1958, Document No. 29.
7. R. L. Franks et al., "A Model Relating Measurements and Forecast Errors to the Provisioning of Direct Final Trunk Groups," B.S.T.J., *58*, No. 2 (February, 1979), pp. 351–77.
8. D. W. Hill and S. R. Neal, "Traffic Capacity of a Probability-Engineered Trunk group," B.S.T.J., *55*, No. 7 (September, 1976), pp. 831–42.

9. C. R. Szelag, "Trunk Demand Servicing in the Presence of Measurement Uncertainty," B.S.T.J., *59,* No. 6 (July–August, 1980), pp. 845–60.
10. V. S. Mummert, "Network Management and its Implementation on the No. 4 ESS," Int. Switching Symp., Japan, 1976.
11. P. R. S. Kumar and K. S. Narendra, "Learning Algorithm Model for Routing in Telephone Networks," Systems and Information Sciences Report No. 7903, Yale University, May, 1979.
12. K. S. Narendra and M. A. L. Thathachar, "On the Behavior of a Learning Automaton in a Changing Environment with Application to Telephone Traffic Routing," Systems and Information Sciences Report No. 7803, Yale University, October, 1978.
13. D. M. McKenna and K. S. Narendra, "Simulation Study of Telephone Traffic Routing Using Learning Algorithms," Technical Report No. 7806, Yale University, 1978.
14. A. E. Bean and E. Szyhicki, "Advanced Traffic Routing in Local Telephone Networks: Performance of Proposed Call Routing Algorithms," 9th Int. Teletraffic Cong., Torremolinos, Spain, October, 1979.